

WHAT IS CLAIMED IS:

1. A multi-threaded server accept method, comprising:
creating a socket accept thread by a control thread of a server process;
receiving a service request from a client by the socket accept thread;
transferring the request to a data structure; and
retrieving the request, by the control thread, from the data structure;
transferring the request to a client thread, by the control thread, to process
request data associated with the request; and
processing the request data by the client thread.
2. The method of claim 1, wherein the data structure comprises a queue.
3. The method of claim 1, wherein the data structure comprises a FIFO
queue.
4. The method of claim 1, further comprising waiting for service requests
by performing an accept () call.
5. The method of claim 1, wherein receiving the request comprises
receiving a client socket object.
6. The method of claim 1, further comprising waiting for the service
request from the client by the socket accept thread.
7. The method of claim 1, further comprising creating the client thread to
process the request data.

8. The method of claim 1, further comprising:

receiving a second request by the socket accept thread from the client;

transferring the second request to the data structure;

retrieving the second request by the control thread;

transferring the second request to a second client thread to process second request data; and

processing the second request data by the second client thread.

9. The method of claim 8, further comprising creating the second client

thread to process the second request data.

10. The method of claim 1, wherein the socket accept thread and the control thread are executed on a single processor.

11. The method of claim 1, wherein the steps of transferring the request to the data structure and retrieving the request from the data structure are serially performed.

12. A multi-thread server accept system, comprising:

a server process residing on a server and operable to

create a socket accept thread by a control thread of a server process residing on the server;

receive a service request from a client by the socket accept thread;

transfer the request to a data structure;

retrieve the request, by the control thread, from the data structure;

transfer the request to a client thread, by the control thread, to process request data associated with the request; and

process the request data by the client thread.

13. The system of claim 12, wherein the data structure comprises a queue.

14. The system of claim 12, wherein the data structure comprises a FIFO queue.

15. The system of claim 12, wherein the socket accept thread is operable to wait for service requests by performing an accept() call.

16. The system of claim 12, wherein the socket accept thread is operable to receive the request by receiving a client socket object from the client.

5 17. The system of claim 12, wherein the server process is further operable to create the client thread to process the request data.

18. The system of claim 12, wherein the server process is further operable to:

10 receive a second request from the client by the socket accept thread after transferring the request to the data structure;

transfer the second request to the data structure;

retrieve the second request by the control thread;

transfer the second request to a second client thread to process the second request data; and

15 process the second request data by the second client thread.

19. The system of claim 12, wherein the server process is further operable to create the second client thread to process the second request data.

20. The system of claim 12, wherein the socket accept thread and the control thread are executed on a single processor.

20 21. The system of claim 12, wherein the server process is further operable to serially perform the steps of transferring the request to the data structure and retrieving the request from the data structure.

22. A multi-threaded server accept application, comprising
an application software residing on a computer-readable medium and operable
to:

5 create a socket accept thread by a control thread of the application
software;

receive a request from a client by the socket accept thread;

transfer the request to a data structure;

retrieve the request, by the control thread, from the data structure;

10 transfer the request to a client thread, by the control thread, to process
request data associated with the request; and
process the request data by the client thread.

23. The application of claim 22, wherein the data structure comprises a
queue.

15 24. The application of claim 22, wherein the data structure comprises a
FIFO queue.

25. The application of claim 22, wherein the application software is further
operable to wait for service requests by calling an accept() program.

20 26. The application of claim 22, wherein the application is further operable
to receive the request by receiving a client socket object from the client.

27. The application of claim 22, wherein the application software is further
operable to create the client thread to process the request data.

10057135-102901

28. The application of claim 22, wherein the application software is further operable to:

5 receive a second request from the client by the socket accept thread after transferring the request to the data structure;

transfer the second request to the data structure;

retrieve the second request by the control thread;

transfer the second request to a second client thread to process second request data; and

10 process the second request data by the second client thread.

29. The application of claim 22, wherein the socket accept thread and the control thread are executed on a single processor.

30. The application of claim 22, wherein the application software is further operable to serially perform the steps of transferring the request to the data structure and retrieving the request from the data structure.

15

1005725:0001